

# Creating More Maintainable Web Automation

---

IMPROVING READABILITY AND REDUCING DUPLICATION  
WITH PAGE OBJECT MODELS



**Jason Roberts**

.NET MVP

@robertsjason    dontcodetired.com



# Overview



Introduction to page object models

Insulate test code from UI changes

Reduce test code duplication

Improve readability of test code

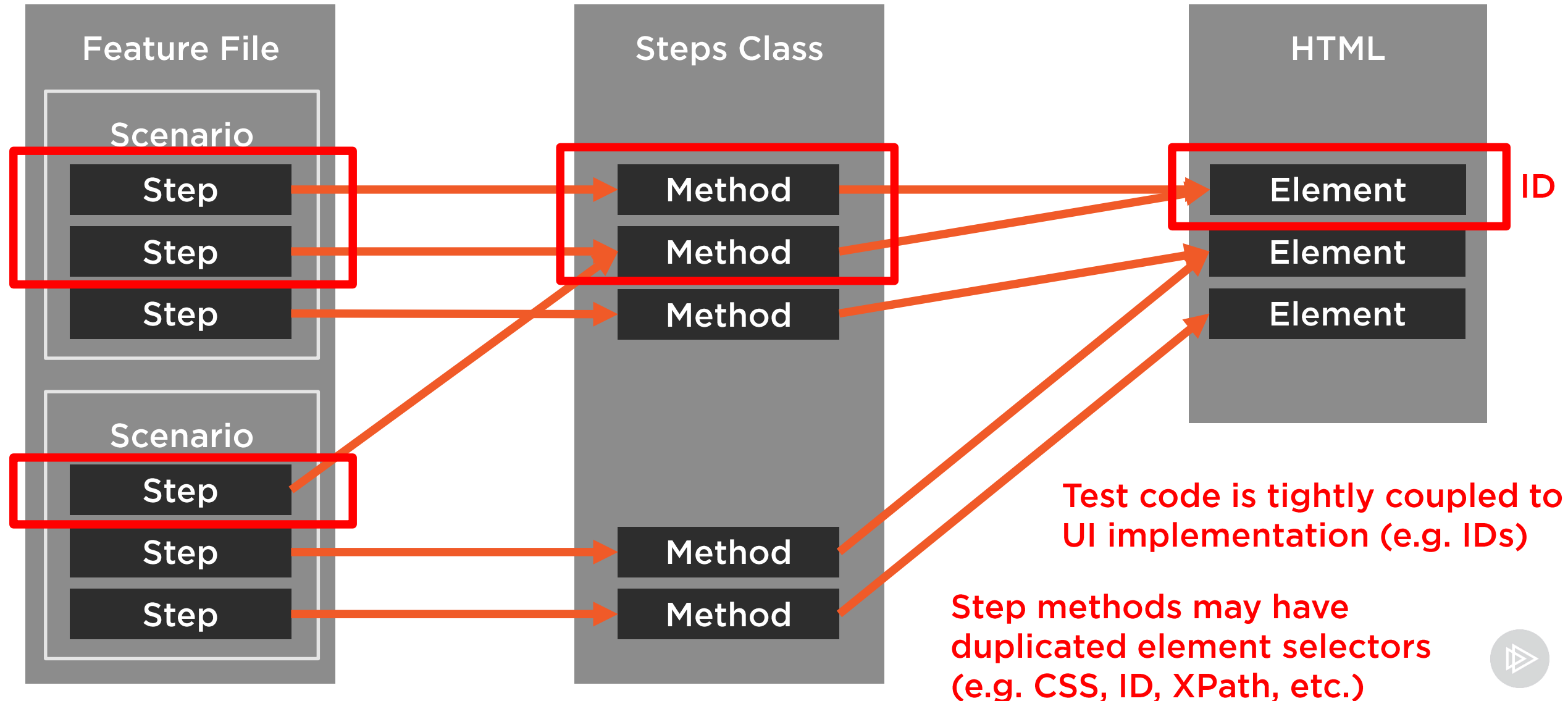
Create page object models for loan application and confirmation pages

Refactor steps code

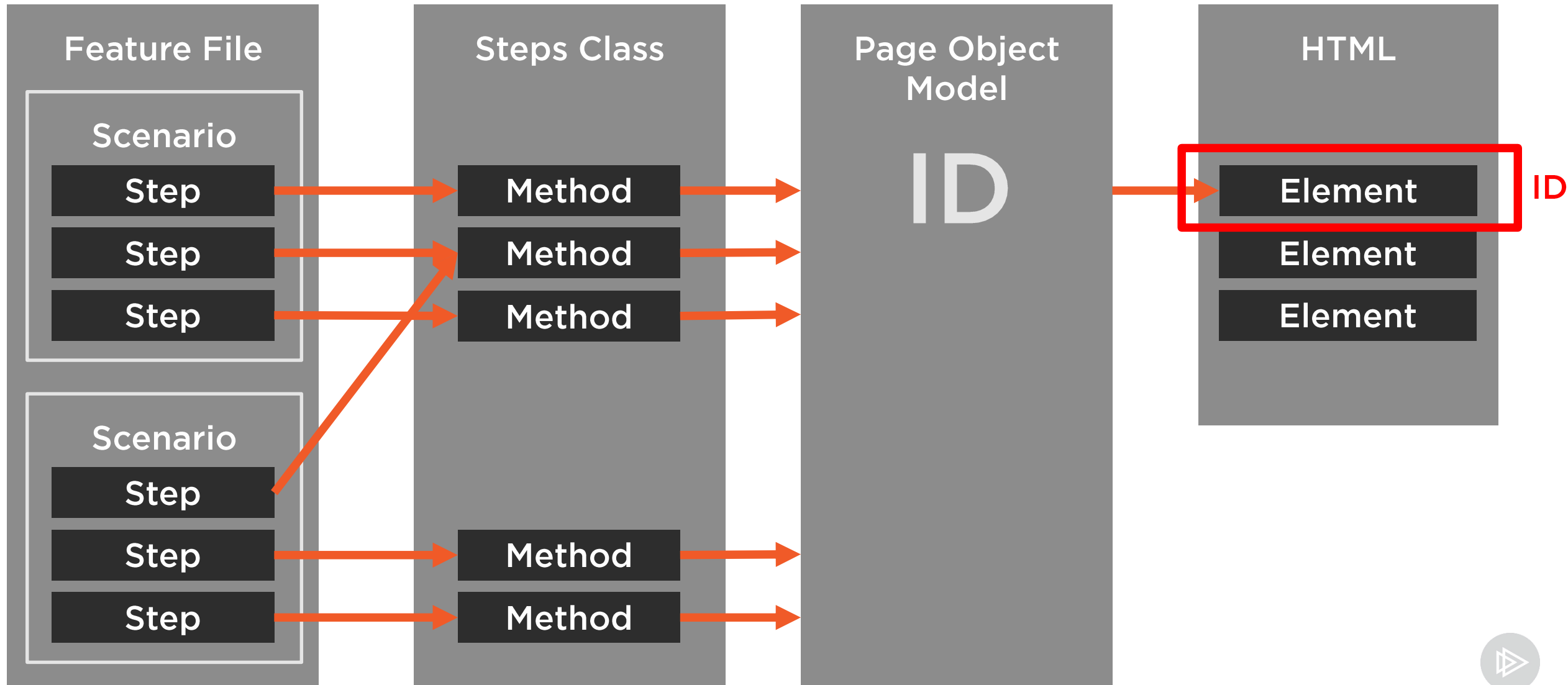
Additional Selenium page object model support



# Introducing Page Object Models



# Introducing Page Object Models



# Improved Readability

```
_driver.FindElement(By.Id("LastName")).SendKeys(secondName);  
_loanApplicationPage.SecondName = secondName;  
  
_driver.FindElement(By.Name("TermsAcceptance")).Click();  
_loanApplicationPage.AcceptTermsAndConditions();  
  
_driver.FindElement(By.CssSelector("div.validation-summary-  
errors ul li"));  
_loanApplicationPage.ErrorText
```



# Demo



Creating a Loan  
Application Page  
Object Model

New LoanApplicationPage class

IWebDriver in constructor

Static NavigateTo method

Setter properties (SendKeys)

Getter properties (.Text)

Action methods:

- SelectExistingLoan()
- AcceptTermsAndConditions()
- SubmitApplication()

***C# 6.0 expression bodied methods and  
get-only properties***



# Demo



Creating an  
Application  
Confirmation Page  
Object Model

New `ApplicationConfirmationPage` class

`WebDriver` in constructor

Getter property

Confirmation first name



# Demo



Refactoring Steps  
to Use Page  
Object Models

Add LoanApplicationPage and  
ApplicationConfirmationPage fields

Modify step class methods

```
_driver.FindElement(By.Id("LastName"))  
    .SendKeys(secondName);
```

```
_loanApplicationPage.SecondName  
    = secondName;
```

Run both scenarios





# Demo



Refactoring to  
Selenium Page  
Object Models

Install Selenium.Support NuGet

Add private fields decorated with  
[FindsBy(...)]

PageFactory.InitElements(\_driver, this);

Reference fields from methods and  
properties

Run both scenarios



# Summary



Introduction to page object models

Insulate test code from UI changes (e.g. HTML element ID changes)

Reduce test code duplication

Improve readability of test code

Created LoanApplicationPage and ApplicationConfirmationPage classes

Ctor with IWebDriver parameter

Added properties & methods

Refactored steps code

[FindBy(...)]





## Resources and Further Learning

### SpecFlow

- [www.specflow.org](http://www.specflow.org)
- “Business Readable Automated Tests with SpecFlow 2” Pluralsight course

### Selenium

- [www.seleniumhq.org](http://www.seleniumhq.org)
- [github.com/SeleniumHQ](https://github.com/SeleniumHQ)

### Seleno

- “Automating Web Applications with Seleno” module
- “Building the Right Thing in .NET with TestStack”

